

XVII. Magyar Számítógépes Nyelvészeti Konferencia Szeged, 2021. január 28–29.

Egy szabály- és következtetésvezérelt csevegőrobot modell anatómiája

Az ITSy-Bitsy modell

Kilián Imre

Deutsche Telekom Systems Solutions Hungary Kft.
7621 Pécs, Hungary
Imre-Zoltan.Kilian@t-systems.com
tel: +49 69 9731792500
www.deutschetelekomitsolutions.hu

Kivonat Csevegőrobotok az elmúlt években robbanásszerűen terjedtek el az ipari szoftverekben, megvalósítva egy természetes nyelvű csevegést lehetővé tevő, újszerű kezelői felületet. Az ipari környezetben használt csevegőrobotok neurális hálós/mélytanulási technológiával készülnek. Itt most mégis egy szabály- és következtetéses alapon készült csevegőrobot modellről számolunk be, és részletezzük a nyelvi feldolgozás menetrendjét, különös tekintettel a tartalmi elemző és átalakító modulra. A cikk végén összehasonlítást teszünk a mélytanulási technológiával elérhető képességek között.

1 Bevezetés

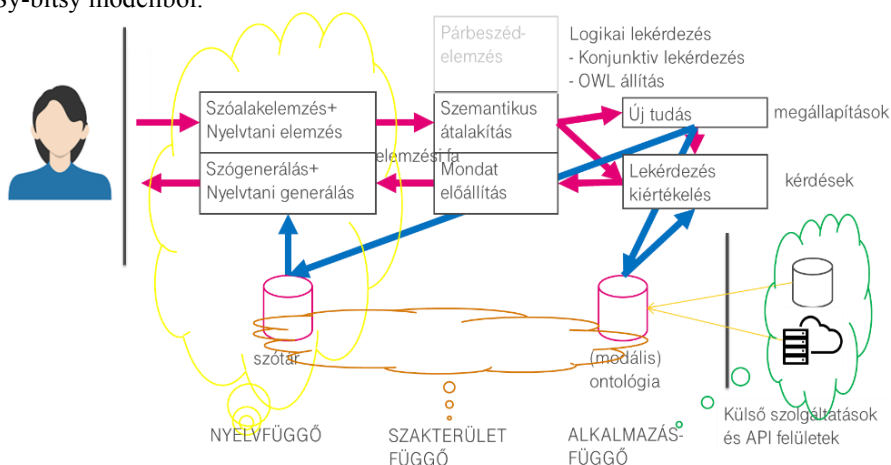
A *csevegőrobotok* (*chatbotok*) nem kevesebbet, mint az Alain Turing kitűzte célt követik: olyan gépi eszközöket valósítanak meg, amelyek természetes nyelvű párbeszéd (csevegés) keresztül érnek el valamilyen célt – amit a háttérben működő alkalmazói program valósít meg (Corydon 2012). A modern robotok képességei ezért általában valamiféle kötött tématerületre vannak szabva. A csevegőrobot programozás alapvető gyakorlata pl. egy Gyakran Ismételt Kérdések (GYIK) tudásanyag betanítása.

Az 1966-ban Joseph Weizenbaum által készített ELIZA robot (Weizenbaum 1966) az első efféle volt. A robot egy pszichiáter szerepét játszotta, és a páciens begépelte mondataiból szövegszerű átalakításokkal kérdéseket csinált úgy, hogy az átalakítási mintáiból véletlenszerűen választott egyet. Az angol nyelv különösen alkalmas az „I am” → „You are” jellegű átalakításokra. Például, ha a páciens a következő mondatot gépelte be: „I am deprived today.”, arra a következő választ (vagy valami hasonlót) adhatott: „Why are you deprived today?”

Az iparban használt csevegőrobotok jellemző megoldása: nagyobb cégeknek saját csevegőmotorjuk van, amit neurális háló valósít meg. Ezek többcélúan testre szabhatók, a szókészletük, a felismert mondatmintáik, sőt, a párbeszéd mintáik programozhatók, a felismerést végző hálózat betanítható, és az adatbázisokba vagy szolgáltatásokba történő lekérdezésük pedig konfigurálható. (Watson, Rasa). Az alább leírt megoldás az

ITSy-bitsy, szabály- és következtetésvezérelt csevegőrobot felépítését tárgyalja, amely Prolog programkörnyezetben született.

A nyelvfeldolgozás klasszikus menetrendjét Vauquois óta ismerjük [4], az ITSy-bitsy szintén ezt az elvet követi. A feldolgozás a bemenő szöveg szimbólumokra tördelésével kezdődik, amit a szóalakelemző egy absztrakt szóalak-szerkezetté alakít át. Ez a nyelvtani elemző bemenete, amely a mondatból egy elemzési fát hoz létre. A nyelvtani elemzés feletti réteg a lehet a párbeszédelemzés, ilyen lehetőséget az ITSy-bitsy jelenleg nem nyújt. A mondatok (bekezdések) elemzési fájából indul ki a tartalmi elemzés, ami egy kiértékelésre, ill. végrehajtásra kész logikai alakot bocsát ki magából. Az ezen réteg feletti pragmatikus szint a szándékok és lehetőségek, netán metakommunikációs eszközök figyelembevételével árnyalja a megértést. Ez a réteg szintén hiányzik az ITSy-bitsy modellből.



1. ábra Az ITSy-bitsy felépítése a klasszikus elveket követi

2 Nyelvi szint

2.1 Nyelvi elemzés

A nyelvi elemző angol nyelvre lett megvalósítva, amelyben a szóalakelemzés feladata leegyszerűsödik, és a kötött szórend a nyelvtani elemzést is megkönnyíti.

A szóalakelemzés persze nem tűnik teljesen el. Mivel formailag az *s* igerag a többes számtól nem különíthető el, és több szó egyidejűleg többféle szófajként is elemezhető, már a szóalakelemzőnek is *nemdeterminisztikusan* kell működnie. (Pl. a 'works' szóalak egyaránt elemezhető igeiként és többes számú főnévként.)

A nyelvtani elemző alulról-felfelé irányban, szintén nemdeterminisztikusan működik, lényegileg a Cock-Younger-Kasami elemzőalgoritmus Prologra alkalmazott, *mélyégi, visszalépéses keresési stratégiával* megvalósított változata. Az elemző

Contralogra épül, ami a Prologgal illeszkedő, ahhoz készített, előrehaladó módon következtető előfordító (Kilián, 2016).

Az alulról felfelé működésmód miatt az elemző a szavak és kifejezések összes lehetséges elemzését előállítja. Ezek közül egyesek a további elemzési lépések során elhalnak, mások viszont részt vesznek a mondat teljes elemzési fájának felépítésében. Ennek során két gond merülhet fel. Egyrészt az elhaló elemzési ágak előállítása felesleges, de ezt korábban még nem mindig tudjuk eldönteni. Másrészt az elemző alternatív elemzések tömegét állíthatja elő, amiről esetleg nem is tudjuk, hogy melyiket fogadjuk el belőlük érvényesnek. Persze léteznek tényleg többféleképpen elemezhető (és értelmezhető) mondatok is, ilyenkor a többszörösség kezelése (nemdeterminizmus) nemigen megtakarítható.

Az elemzések előállítása során az alternatív elemzések nem látnak át egymás részeredményeire, ezért nem tudunk különféle heurisztikus kikötéseket tenni. (pl., hogy egy adott fa-mintára illeszkedően a lehető legnagyobb szövegszeletet fogadjuk el, ill. hogy egy adott szövegszeletre illeszkedő lehető legegyszerűbb fa legyen a nyerő.)

Nyelvtani elemzőnk kimenete az elemzett mondat *elemzési fája*. Ez egyrészt a Prolog megvalósítási nyelven természetes módon ábrázolható, másrészt az elemzési fából *szöveget generáló* modul nem csinál mást, mint az elemzési fát járja körül.

Az ITSy-bitsy a jelen állapotában az A2 (erős kezdő) szintre van beprogramozva. A szótárszerkezet bővíthető, történt már sikeres kísérlet a B2 (erős haladó) szint elérésére, mindenesetre nyelvtani bővítés nélkül. A szótárbővítés lehetővé teszi szakterület-függő szókészlet beépítését is. Végso célként a C (anyanyelvi) szint is kitűzhető, de félo, hogy a szaporodó intuitív megértési problémák miatt nemigen érhető el.

2.2 A nyelv határai

Az elemzési alapfeladat egyes – jól formált – mondatokat elfogad, másokat elvet. Készíthető olyan elemző, amely az akadémiai nyelvet elfogadja, a többit viszont elveti. Elvetheti pl. egyes területileg, szubkultúrák, életkorok, netán foglalkozások által meghatározott nyelvjárások mondatait. Másrészt viszont akadémiai nyelv nem létezik, mert azt a legjobb esetben is csak írásban használjuk – még a legszabatosabb beszélő mondatainak is csak kis része fér bele az akadémia keretekbe. Sejtésünk, hogy az akadémia nyelvet bármilyen tökéletesen elemesse egy program, az már a ténylegesen használt írott nyelv vizsgáján is megbukik, a beszélt nyelvről nem is beszélve.

3 Tartalmi elemzés

A tartalmi elemzések alapfeltételezése, hogy a mondatok logikai alakját a háttérben, *fesztes logikai alakban tárolt, és hétköznapi fogalmakat tartalmazó tudásállományhoz* illesztjük, és megállapítjuk, hogy következhet-e abból, netán ellentmondásban áll-e vele. A háttérben tárolt hétköznapi tudást régebben *tudásbázisnak*, manapság inkább *ontológiának* nevezik.

3.1 Statikus szemantika

Nyelvtanilag jól formált mondatokkal is lehet értelmetlen dolgokat mondani (...mert mi lenne akkor a költőkkel? /Bach Iván/). Ezt az ontológia modellszintje alapján az elemzett mondatok *statikus szemantikus* ellenőrzésével ellenőrizhetjük, ami egyes felesleges elemzési tévutak szűrésére is használható. (pl. „Színtelen zöld eszmék dühödten alsznak.” /Chomsky/)

Ha ugyanis a modellszinten az egyes relációk értelmezési tartományát is tároljuk, akkor a fenti mondat pl. a következő szemantikai szabályokat sérti:

$\text{Dom}(\text{Verb}(\text{sleep})) \cap \text{Concept}(\text{idea}) = \emptyset$ az 'aludni' ige értelmezési tartományának és az 'eszme' fogalomnak nincs közös része.

$\text{Dom}(\text{Adj}(\text{green})) \cap \text{Concept}(\text{idea}) = \emptyset$ a 'zöld' jelző értelmezési tartományának és az 'eszme' fogalomnak nincs közös része.

$\text{colorless} \cap \text{colorful} \supset \text{green} = \emptyset$ a 'színes' és a 'színtelen' egymás kiegészítő halmazai, 'zöld' részhalmazzal nincs közös része.

3.2 Átalakítás logikai nyelvvé

Az átalakítás során a mondat elemzési fájából egy *logikai nyelvű mondatot* hozunk létre. Alapvető kérdés a logikai nyelv meghatározása: Nagyon magas szintű logikai nyelvnek nagy a kifejező ereje, ám ezek a nyelvek legtöbbször *eldönthetetlenek*. Alacsonyabb szintű nyelvek annyira szószátyárok, hogy a gyakorlatban sajnos használhatatlanok is. Emellett fontos szempont lehet a nyelvhez készített megoldó/kiértékelő szoftver csomag elérhetősége. Ebből a szempontból a Prolog ideálisnak tűnik.

A logikai alakba történő átalakítás nagymértékben függ a mondatban használt *igemódtól*.

A *feltételes módtól* ezúttal eltekintünk, mert az az angol nyelven csupán feltételes kötőszóval összekapcsolt, és időbeli viszonyba állított kijelentő tagmondatokkal van megvalósítva.

Felszólító mondatokat a csevegőrobotok valami azonnali tevékenységre utaló parancsként értelmezhetnek. Közvetlen tevékenység végrehajtásának különböző ügyfélszolgálatok esetén sok jelentősége lehet, de az irodai munka is tartogat efféle lehetőségeket.

Különösen érdekes a *kijelentő és a kérdő mondatok* megvalósítása. A kérdő mondatokat jelenleg az elsőrendű logikai modellben ún. *konjunktív kérdésekké*, vagyis elemi relációk konjunkciójává alakíthatjuk át, amelyet a Prolog alapú tudáskezelő rendszer értékel ki és válaszol meg.

A kijelentő mondatokat viszont aszimmetrikus módon OWL (Web Ontology Language) állításokká képezzük le, majd a tudásállományhoz hozzávesszük.

3.3 Kétszintű logikai modell

A hétköznapi élet mondatai és kérdései sajnos nem mindig elsődrendűek. Az efféle problémák kezelését ún. *kétszintű tudásábrázolással* oldhatjuk meg, amit *reifikációnak* is neveznek. A *példányszintű / adatszintű* tudáselemeken (pl. Kala Pál mikor hol, mit csinált) kívül *modellszintű* tudáselemeket (tudásszegmenst) is tartalmaz, ami a példányszinten használt modelleszközökre: osztályokra, tulajdonságokra és relációkra vonatkozó általános ismereteket, pl. az értelmezési tartományait, ill. azok egyéb összefüggéseit tárolja.

A kétszintű tudásábrázolás már másodrendű kérdésekre is választ adhat. Pl. a „Mi a különbség Malacka és Bagoj (sic!) között?” kérdésből egy olyan logikai kifejezést hozhatunk létre, amely egyrészt modellszinten megkeresheti a két állatkára csak külön-külön vonatkozó relációkat, de a közös tulajdonságok és relációk különböző értékeit is megkeresheti. Például a *hasKeeper/vanGazdája* tulajdonság értéke mindkettőjükre „Christopher Robin”, míg a *hasResidence* reláció értéke különböző, a *nestingTime/költésiIdő* reláció viszont Malackára, mint emlőslátra nincs is értelmezve.

3.4 Modális logikai modell

Logikai alapú rendszerek egyik gondja az *ellentmondás-mentesség*. Effélék nem modellezhetők, pedig a valóságos élet tele van ellentmondásokkal. Ellentmondások kezelésének egyik útja a *modális logikák* bevezetése, ahol *világocskákat (környezeteket)* határozunk meg. Az ellentmondás-mentesség csak egy világocskán belül követelmény, a világok között már nem.

A világocskák persze lehetnek *halmazok* is. A jogi normatívák pl. az ún. *deontikus* világocská-halmazban értelmezhetők, amely a TILOS, MEGENGEDETT, VÁLASZTHATÓ, KÖTELEZŐ világocskákat (operátorokat) tartalmazza. Az egyes világocskák között a következtetések végzéséhez különféle logikai axiómákat szokásos megállapítani.

Világocskákat az *episztemikus-doxasztikus* logika alapján *hozzárendelhetünk gondolkodó ügynökökhöz (ágensekhez)* is, pl. *Know*(WinnieThePooh), *Believe*(Piglet). A világocskák egymásba is ágyazhatók, pl. *B*(Piglet, *K*(Heffalump)). Az egymásba ágyazott világocskák között gyakori az *öröklés*, mint logikai axióma használata, a világocskák globális gyökéreleme ilyenkor az általánosan elfogadott/megdönthetetlen állításokat tartalmazó világ (Alberti 2009, Kilián 2012).

A világocskákat operátorként értelmezve a logikai kifejezések elé írjuk, pl: *K*(WinnieThePooh) (*HasLittleBrain*(Ego) , *Bear*(Ego) , *Child*(Ego))

... vagyis: Micimackó tudja magáról, hogy ő egy csekélyértelmű medvebocs.

A mondatok időparaméterét – mint a mondat egészére vonatkozó információt – szintén felfoghatjuk modális környezetként, amit külön dimenzióként együtt alkalmazhatunk az egyéb világocskákkal. Személetes ábrázolásmódban az egyéb világocskák „szőreiként”, vagyis beágyazott apró *levél-világocskákként* érdemes őket ábrázolni, és a következtetések elvégzésére itt is rögzíthetők különféle axiómák (Ruzsa, 1984) Időlogikában a következő operátorok használatosak:

BeforeSometime(T) a T időpont előtt valamikor igaz volt

$\text{BeforeAlways}(T)$	a T időpont előtt mindig igaz volt
$\text{AfterSometime}(T)$	a T időpont után valamikor igaz (lesz)
$\text{AfterAlways}(T)$	a T időpont után mindig igaz (lesz)

Példa axiómákra, amelyek segítségével logikai következtetések végezhetők:

$W_{\text{Off}} \phi \leftarrow W_{\text{Anc}} \phi$	a világcskák közötti tudásöröklődés axiómája
$K(WHO) \phi \rightarrow B(WHO) \phi$	episztemikus fokozatöröklődés axiómája
$AS\phi \vee AS\psi \rightarrow AS(\phi \vee AS\psi)$	
$\vee AS(AS\phi \vee \psi)$	
$\vee AS(AS\phi \vee AS\psi)$	ha a jövőben valamikor egy dolog vagy egy másik igaz lesz, akkor vagy az egyik igaz és a másik még nem, vagy fordítva, vagy még egyik sem igaz

3.5 Átalakítási szabályok

Az alábbiakban állítások és kérdések konjunktív logikai lekérdezőnyelvvé történő átalakítását mutatjuk be elsődrendű logika felett.

Igei kijelentő mondat modellje elsődrendű nyelven:

$\text{alany}(\dots), \text{argumentumok}(\dots) \rightarrow \text{ige}(\dots)$

Névszói kijelentő mondatok esetében az állítmány nem ige, hanem névszó, tehát megfelelő logikai kifejezést is a névszói szerkezetekhez hasonlóan képezzük. *Névszói szerkezetek* átalakítási mintája az ún. *összegző (kumulatív)* értelmezés szerint:

$\text{jelző}_1(X), \dots, \text{jelző}_N(X), \text{köznév}(X)$

Tulajdonnevek egy egyedi azonosítóba képződnek le. Az azonosító az öt tartalmazó legszűkebb osztályon belül egyértelmű, vagyis az azonosítót az osztálynévvel is címkézzük. Pl. a „Malacka a Százholdas Pagonyban lakik.” állításból az alábbi logikai (tény) -állítás keletkezik.

$\text{live}(\text{piglet}\#\text{pig}, \text{hundredAcreWood}\#\text{woodland}).$

Az általános modellhez képest természetesen kivételek is megfogalmazhatók, pl. a „szétszórt bölcsészlány” kifejezés inkább önmagában is egy következtetés ($\text{szétszórt}(X) \leftarrow \text{lány}(X), \text{bölcsész}(X)$) (Alberti, 2011) Emellett az igei kezelési módja sem mindig egyöntetű.

- *Tranzitív ige*k bináris, *ditranzitív ige*k három oldalú relációkba fordíthatók, és ezek sorrendje rögzített ugyanúgy, mint a kötelező vonzatoké is.
- Egy-egy *ige egyéb paramétereit* azonban vonzat helyett gyakran *szabad határozókként* ismeri fel az elemző. Az ontológia felépítésétől függően ezeket gyakran újabb relációparaméterként kell beszúrni.
- Egyes *általános jelentésű ige*k relációs fordítását újabb *típus paraméterrel* is kiegészítjük. Tipikusan ilyen a *have* ige, amelyet ilyen módon a birtoktárgy típusától függően részrelációkra bontunk.

Kijelentő mondatok átalakítása tényállításokat eredményez. Ha az alany és az esetleges igei argumentumok leírása bonyolultabb (pl. kvantorokat, birtokviszonyt stb. is tartalmaz), akkor a logikai alak egy Horn-klóz, vagyis egy Prolog szabály lesz. Attól függően, hogy az új tényállítás vagy szabály már létezik-e, döntünk annak felvételéről. További kérdés, hogy *melyik világocskába* vegyük fel az új ismeretet, amit stratégiai megfontolásokkal dönthetünk el. Egy *bizalmatlan* stratégia (pl. egy bíró a bíróságon) minden információt a kijelentést tevő egyén világocskájába helyez, míg egy *hiszékeny* stratégia mindent a gyökérvilágba, vagyis a megdönthetetlen állítások közé.

Eldöntendő (yes/no) kérdő mondatok esetében a mondat logikai alakját kiértékeljük, és a logikai eredményt visszaadjuk.

Kiegészítendő kérdésekben a kérdőszó vagy kifejezés gyakran egy igei vonzat szerepét játssza. A kérdőszóból logikai változó lesz, az egész mondat pedig olyan logikai kifejezéssé fordul le, amely ezt tartalmazza. A kérdés kiértékelése során a változó értéket kap, ez lesz a kérdésre adandó válasz. A kiegészítendő kérdés modellje tehát:

$X^{\text{mondat}}(X)$.

Kvantorok az elemzett nyelvben előfordulhatnak explicit módon is, de több ösztönös vagy rejtett nyelvi kifejezés is gyakran csak kvantorokkal (*gyűjtőfüggvényekkel*) értelmezhető. Jelenleg a következő gyűjtőfüggvények kezelése van megvalósítva.

- Egyes számú, de a nem konkrét főnévi csoport esetén a kérdésből létrehozott egyszerű kérdés kerül végrehajtásra. Ha az ismeretlennek több lehetséges értéke is volna, akkor visszalépéssel előállítja őket, pl. a „Who is Baby Roo’s male friend?” kérdésre válaszolva.

```
X:=X^(male(X), friend(X),
      have(babyRoo#kangaroo, friend, X))
```

- Többes számú főnévi csoport esetén az eredmény egy halmaz. Az ilyen kérdésből létrehozott lekérdezés (ld. alább) a COND feltételnek megfelelő X változókat gyűjti össze a LIST változóba, pl. a „Who are the male friends of Piglet?” feltevése esetén.

```
LIST:=each(X^male(X), friend(X),
           have(babyRoo#kangaroo, friend, X))
```

- Ha határozott névelőt használunk, akkor arra utalunk, hogy a körülírt dologból egyetlenegy létezik. Az ilyen kérdésből létrehozott lekérdezés az *egyetlen, a feltételeknek megfelelő változóértéket* adja vissza.

```
VALUE:=only(X^COND)
```

- Gyakran valamilyen a feltételnek megfelelő *gyűjtemény számosságára* kérdezzünk. Ez az alábbi lekérdezéssel lehetséges:

```
NR:=count(X^COND)
```

- *Helyhatározói kérdésekre* van egy további érdekes beépített feldolgozás. Ha ugyanis több szereplőre vonatkozólag valami helyhatározós kérdés tennénk fel, (pl., hogy hol laknak), akkor a földrajzi egész-rész reláción (hasGeopart/2) keresztül a legközelebbi közös lakóhelyet kapjuk válaszul (lca: least common ancestor). A kérdés általános alakjában paraméterül adjuk a relációt magát is, ld. alább.

$X := \text{lca}(X^{\text{CALL}}, \text{RELATION})$.

Például, a „Where do Piglet and Owl live?” kérdésre a „HundredAcreWood” választ kapjuk, míg a „Where do Kanga and Baby Roo live?” kérdésre a kangasHouse a helyes válasz. (Az eltérő írásmód azt jelzi: egy azonosítóról, és nem valami-féle kifejezésről van szó, amelynek nyelvtani kifejtése nincs pontosabban meghatározva.)

3.6 Kapcsolat konkrét szolgáltatásokhoz

A szoftver tartalmaz egy csatolómodult, amelyben a konkrét Prolog ontológiához történő kapcsolódás van leírva: vagyis esetleges Prolog átalakító-szabályokat lehet/kell itt megadni. Külső adatbázisok vagy szolgáltatások esetén meg kell valósítani az ontológiát kiterjesztő műveleteket is (vagyis ami lekérdez, bővít, töröl egy olyan tudáselemet, amely külső eszközön vagy szoftveren áll rendelkezésre).

4 Összehasonlítás mélytanulásos megoldásokkal

A napjainkban elterjedt mélytanulásos megoldások sikere kikényszeríti, hogy egy szabályalapú megoldás készítője legalábbis áttekinthesse, kiértékelje egy mélytanulásos eszköz működését, és – ha nem is pontos mérési módszereken keresztül – de legalább hevenyészett összehasonlítást tegyen.

Heading level	Szabályvezérelt	Mélytanulásos
Tanítás	kevesebb szabállyal	sok példával
Működés	hajszálpontos	valószínűségi
Be nem tanított esetek	intoleráns	toleráns
Következtetés	van	korlátozott
Metaszintek	tudja	nem tudja
	hasonlít az anyanyelv-használathoz	hasonlít az idegen nyelv használatához

2. ábra Szabályvezérelt és a mélytanulásos megoldások összehasonlítása

Az összehasonlításból végeredményben az derül ki, hogy a következtetésvezérelt megoldás az általa szállított megoldásokra hajszálpontos, viszont be nem tanított megoldásokra intoleráns. Talán kicsit az anyanyelvi beszélő nyelvértéséhez hasonlít, aki általában a közölt mondat minden egyes szavát pontosan érti, és pontosan meg tudja

válaszolni, de esetleg egy akár tájnyelven használt, nem ismert vagy oda nem illő kötő- vagy indulatszó komolyan megzavarhatja a megértést.

A mélytanulós megoldás talán inkább az idegen nyelvek megértését mintázzák. Ha egy-két szó ismeretlen is egy mondatban, attól még a lényegét megértjük, és közben persze bizakodunk, hogy a fel nem ismert szavak nem változtatják meg gyökeresen a mondat értelmét.

Talán az intelligens beszélő stratégiája a legszerencsésebb: ha pl. egy csángó nyelv-járású adatközlő minden szavát nem érti hajszálpontosan, stratégiát vált, és valószínűségi alapon, valamint egyes metakommunikációs jelekből fogja a mondanivalót. Az efféle megoldásokat a műszaki életben *hibridnek* nevezik.

5 Értékelés és további munkák

Az ITSy-bitsy a jelen állapotában egy állapotmentes csevegőrobot, amely angol nyelven működik. A moduláris felépítésből következőleg más elemző vagy logikai alakra hozó modul is bekapcsolható, és más logikában másféle kiértékelés is használható. Annak sincs akadálya, hogy valamelyik modult másféle technológiára (pl. mélytanulósra) cseréljük (Wei-Rui, 2018).

A robot az SWI-Prolog rendszer környezetében készült (Wielemaker, 2003). A rendszerrel lényegileg végigjártuk a csevegőrobot készítés főutcaját. Létrejött egy műszakilag teljes értékű modell, és minden fontos technológiai lépésre kiterjedő feldolgozási lánc, ami jelenleg a kiegészítendő kérdések feldolgozását tudja a legtokéletebben elvégezni (és megválaszolni).

A cikkben leírt kezdeményezés ígéretes, de a további életét teljesen biztosan befolyásolja a *külső érdeklődés*: az a segítség, amivel legalább egy életszerű és életszagú mintaalkalmazás összehozható. Természetesen – magyar környezetben – eléggé fontos volna a magyar elemző létrehozása és beillesztése is...

Hivatkozások

- Alberti, G. 2009. ReALIS: An Interpretation System which is Reciprocal and Lifelong. Workshop 'Focus on Discourse and Context-Dependence' UvA, Amsterdam Center for Language and Comm., 2009.
- Alberti, G.: ReALIS: Interpretálók a világban, világok az interpretálóban. Akadémiai Kiadó, Budapest (2011)
- Corydon, I.: Alain Turing at 100, The Harvard Gazette, Harvard University, September, 2012, (<https://news.harvard.edu/gazette/story/2012/09/alan-turing-at-100>, elérés: 21-07-2020)
- Kilián, I.: ReALIS: egy többszereplős, episztemikus rendszer Prolog modellje. In: SzámOkt 2012. konferencia kiadványa, pp. 276–281, Erdélyi Magyar Műszaki Tudományos Társaság, Kolozsvár (2012)
- Kilián, I.: Contralog: a Prolog conform forward-chaining environment and its application for dynamic programming and natural language parsing, In: Acta Universitatis Sapientiae, 8-1, pp.41-62. Marosvásárhely, 2016.
- Ruzsa, I.: Klasszikus, modális és intenzionális logika. Akadémiai Kiadó, Budapest (1984)

- Vauquois, B. (1968). A survey of formal grammars and algorithms for recognition and transformation in mechanical translation. In: IFIP Congress (2) pp. 1114-1122
- Wei W., Rui, Y.: Deep Chit-Chat: Deep Learning for Chatbots. Conference on Empirical Methods in Natural Language Processing, Brussels, (2018) <http://www.ruiyan.me/pubs/tutorial-emnlp18.pdf>, elérés: 05-12-2020)
- Wielemaker J.: An overview of the SWI-Prolog programming environment. In: Proc. 13-th International Workshop on Logic Programming Environments, pp.1-16. eds: F. Mesnard, A. Serebenik, Katholieke Universiteit, Leuven, Belgium (2003)
- Weizenbaum, J.: ELIZA--A Computer Program for the Study of Natural Language Communication Between Man and Machine (http://www.universelle-automation.de/1966_Boston.pdf, elérés: 21-07-2020).